This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problems Mailbox.

THIS PAGE BLANK (USPTO)

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 285 310**
A2

(12) EUROPEAN PATENT APPLICATION

(21) Application number: 88302516.5

(51) Int. Cl.⁴ G06F 9/46

(22) Date of filing: 22.03.88

(30) Priority: 31.03.87 JP 78506/87

(43) Date of publication of application:
05.10.88 Bulletin 88/40

(84) Designated Contracting States:
DE FR GB

(71) Applicant: KABUSHIKI KAISHA TOSHIBA
72, Horikawa-cho Saiwai-ku
Kawasaki-shi Kanagawa-ken 210(JP)

(72) Inventor: Matoba, Tsukasa c/o Patent Division
Kabushiki Kaisha Toshiba 1-1 Shibaura 1-chome
Minato-ku Tokyo 105(JP)
Inventor: Aikawa, Takeshi c/o Patent Division
Kabushiki Kaisha Toshiba 1-1 Shibaura 1-chome
Minato-ku Tokyo 105(JP)
Inventor: Okamura, Mitsuyoshi c/o Patent Division
Kabushiki Kaisha Toshiba 1-1 Shibaura 1-chome
Minato-ku Tokyo 105(JP)
Inventor: Maeda, Ken-ichi c/o Patent Division
Kabushiki Kaisha Toshiba 1-1 Shibaura 1-chome
Minato-ku Tokyo 105(JP)
Inventor: Saito, Mitsuo c/o Patent Division
Kabushiki Kaisha Toshiba 1-1 Shibaura 1-chome
Minato-ku Tokyo 105(JP)

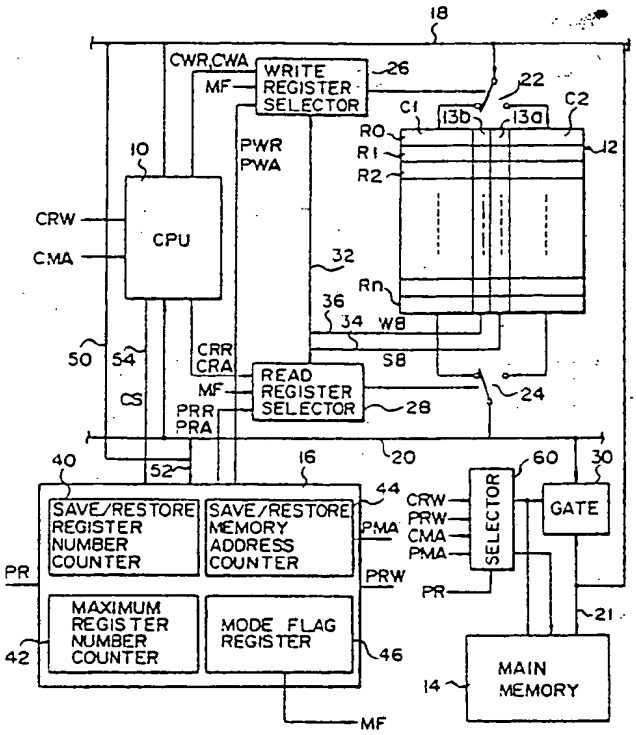
(74) Representative: Freed, Arthur Woolf et al
MARKS & CLERK 57-60 Lincoln's Inn Fields
London WC2A 3LS(GB)

(54) Device for saving and restoring register information.

(57) Each register of an internal register unit (12) of a microprocessor has a pair of register cells consisting of first and second cells (C1, C2) having the same register address. When one of these cells is selected, the other cell non-selected serves as a "back-up cell" for the selected cell. Each register of the register unit (12) has a flag bit (13a) for storing a selector flag (SB) representing which cell of the pair of cells of the register is currently selected, and a flag bit (13b) for storing a change flag (WB) representing whether register information of the register is rewritten after a selected cell is changed between the first and second cells (C1, C2) of the register. When the register information is stored in one of the pair of cells currently being selected of a certain register and is to be rewritten with another new information, the other cell of the register is selected to store the new information therein. The original register information is held in the first cell (C1), thereby eliminating necessity of saving the original register information to a main memory (14) at this stage. When the original register information is required again, the original register information can be rapidly restored, by only selecting the first

EP 0 285 310 A2

cell again. in the corresponding register without executing save/restore processing between the register unit (12) and the main memory (14).



F I G. 1

Device for saving and restoring register information

The present invention relates to a highly-integrated micro-processor and, more particularly, to a device for saving and restoring register information temporarily stored in a general-purpose register of a central-processing unit.

Recently, an architecture of high-performance micro-processors and high-integration fabrication technology thereof have been developed significantly. In particular, the processing capacity of micro-processors has been greatly improved to enable of advanced micro-processors which can provide an information processing environment suitable for a multi-task operating system. Among those micro-processors of this type a micro-processor is known which has, in its chip, instructions for the efficient internal execution of a specific multi-task operating system which enables a single machine to perform a plurality of different information processing (i.e., tasks) in a parallel manner. In the field of special-purpose machines, a microprocessor for a machine (called a "PROLOG machine") which is strongly linked to a "PROLOG", which is a typical one of computer languages suitable for programs for an artificial intelligence, has been widely developed.

In any of the above machines, register contents (e.g., control information, instructions, data and the like are also called as "register information" in this specification) in a micro-processor are frequently changed. For example, assume that a given task is performed on a multi-task operation system and at the same time another task must be performed in a parallel manner. In this case, stored contents of an internal register are changed such that register contents of a current register are saved to a main memory and necessary information are stored at empty register addresses of the register. When performance of the task is finished, the original information saved to the main memory are returned to the register, thereby restoring the register contents. According to the multi-task operation system, process-switching must be performed in units of predetermined time intervals of several milliseconds. Therefore, saving/restoring of processor information concerning the process currently being performed must be frequently and rapidly repeated between the register and the main memory in units of the above time intervals.

In the PROLOG machine, if a so-called "back step" in which a system environment is returned to a state one step before the current state is generated while a unification operation is being performed on an inference program system, all the information concerning the environment up to a closest branch point (called as a "choice point") among a plurality of choice points passed so far must be restored in a register. For this purpose, a large amount of information are transferred between the main memory and the internal register.

In a microscopic view point, when instructions such as a "subroutine call" or "return" included in software programs are to be performed while a given task is being performed, current register contents must be rewritten. In this case, again, current information stored in the register are saved to the main memory, and information required for performing the above instructions are written in the register.

According to current micro-processors, a frequency of saving/restoring of the register contents is significantly increased. Therefore, if the conventional register save/restore technique is adopted, an operation speed of the micro-processor is reduced. The reason for this is as follows. That is, according to the conventional register save/restore technique, each time saving/restoring is to be performed, managements for controlling each section must be performed to execute data accessing in the internal register and the main memory and data transfer therebetween. Therefore, as a time required for managements is increased, a time of a stand-by state for information processing is increased.

Conventionally, in consideration of the above situation, an improved micro-processor has been developed so that a save/restore step of all the register contents in the micro-processor can be performed by only a single instruction. However, according to such a software improvement, although a time required for fetching the instruction in a central processing unit can be reduced, a time required for actually transferring data between the register and the main memory cannot be reduced at all.

It is therefore an object of the present invention to provide a new and improved device for saving and restoring register information in which information transfer time required for saving/restoring register contents between a register and a main memory in a micro-processor is reduced, thereby performing register content save/restore processing at a high speed.

According to the present invention, the above object can be achieved mainly by improving a hardware arrangement of a register unit so that each register in an internal register unit of the micro-processor is constituted by a pair of cells. More specifically, each register has a pair of register cells as first and second cells having the same address. When one of the two cells is selected, the other cell serves as a back-up cell for the selected cell. Each register in the register unit includes a first flag which represents whether the

first or second cell of the register is currently selected, and a second flag which represents whether register information of the register is rewritten after a selected cell is changed between the first and second cells of the register.

A register save/restore controller is connected to a register unit. When the first register information is to be written in a given register of the register unit, the controller selects one of the first and second cells of the register (at this time, the other cell remains as a nonselected cell) into which the first register information is stored. When the first register information is to be rewritten with the second register information, the controller selects the other cell and stores the second register information therein, thereby holding the first register information in one of the cells. Therefore, in this stage, the first register information need not be saved to the main memory. In addition, if the first register information is required after it is rewritten with the second register information, it can be rapidly restored to the corresponding register by only selecting the cell which stores the first register information, without performing save/restore processing between the register unit and the main memory.

The present invention and its objects and advantages will become more apparent in a detailed description of a preferred embodiment to be presented below.

In the detailed description of a preferred embodiment of the invention presented below, reference is made to the accompanying drawings of which:

Fig. 1 is a block diagram showing an internal arrangement of a micro-processor incorporating a device for saving and restoring register information according to a preferred embodiment of the present invention;

Fig. 2 is a schematic diagram showing changes in an execution mode flag which is controlled by the device for saving and restoring register information in Fig. 1 obtained when call mode and return mode events occur during a task performance procedure of a central processing unit; and

Figs. 3A to 7 are schematic diagrams showing how stored contents of a given register of a register unit change when save/restore processing is performed in each of the call and return modes in correspondence to changes in selector and change flags provided in the register.

Referring now to Fig. 1, a micro-processor has central processing unit (to be referred to as a "CPU" hereinafter) 10, internal register unit 12, main memory 14, and register save/restore controller 16 on its chip substrate (not shown). First data bus 18 serves as a data write bus for CPU 10 and register unit 12. Second data bus 20 functions as a data read bus for CPU 10 and register unit 12. CPU 10, register unit 12, and main memory 14 are connected to each other through buses 18 and 20. Therefore, under the control of controller 16, data transfer between register unit 12 and main memory 14 for saving/restoring register contents is performed through buses 18 and 20. Third data bus 21 allows data transfer between main memory 14 and the first and second data buses.

CPU 10 corresponds to a hardware arrangement of a general micro-processor without including a general-purpose register unit: CPU10 is constituted by an arithmetic and logical unit (abbreviated as an "ALU"), a timing and control unit, a program counter, an instruction register, an instruction decoding and machine cycle encoding unit, and the like.

Register unit 12 has an array of (n + 1) registers R0, R1, R2,..... Rn. As shown Fig. 1, each register Ri includes a pair of cells C1 and C2. Cell C1 is called as a "front cell" and cell C2 is called as a "back cell" hereinafter. Front and back cells C1 and C2 have the same address in a memory location of the register, and CPU 10 cannot discriminate the cells.

Each register Ri has flag storage bits 13a and 13b for respectively storing selector flag SB and change flag WB of cells C1 and C2. Selector flag SB is 1-bit binary data representing whether front or back cell C1 or C2 is currently selected in register Ri. Change flag WB is 1-bit binary data representing whether register information currently being used is changed to another new register information. In other words, change flag WB represents whether stored contents of a cell currently being selected are rewritten after the register cell is switched. Change flag WB is reset to have logic value "0" when an event occurs in a call mode such as a subroutine and software interruption and a return mode such as subroutine return and interruption return and a selected cell is changed between cells C1 and C2 of register Ri; and is set to have logic value "1" when the stored contents of the selected cell are rewritten with another information

In other words, register unit 12 is constituted by parallel connection of an array of front cells C1 and that of back cells C2. However, register unit 12 is not a simple parallel connection of these two cell arrays. That is, bits 13a and 13b for storing flags SB and WB which represent current states of front and back cells C1 and C2 are provided for each pair of cells, and a selected cell can be independently switched in units of registers Ri. In other words, a back-up cell is additionally provided to each register address. When one of the two cells of register Ri is currently selected, the other cell serves as a "back-up cell" for the selected cell. This "back-up cell" is indepencently changed each time a selected cell is switched in register Ri. For

example, when front cell C1 is selected in registers R0 and back cell C2 is selected in R1, back cell C2 and front cell C1 serve as "back-up cells" in registers R0 and R1, respectively. In register R1, when a selected cell is switched from back cell C2 to front cell C1, it can be assumed that back cell C2 which is a nonselected cell serves as a "back-up cell" for selected cell C1.

Data write bus 18 is connected to an input terminal of register unit 12 through first data flow controlling switch 22. Data read bus 20 is connected to an output terminal of register unit 12 through second data flow controlling switch 24. Switches 22 and 24 can arbitrarily perform switching between cells C1 and C2 of a desired register of register unit 12 independently of the remaining registers.

Switches 22 and 24 are connected to write register selector 26 and read register selector 28, respectively. When information to be written in register unit 12 is supplied, write register selector 26 appropriately controls switch 22 and sends the information to either of front and back cells C1 and C2 (which is determined by controller 16) at an address (i.e., a register) of a register memory location to be accessed. When contents (register stored information) of register unit 12 are to be temporarily saved to main memory 14 through bus 20, read register selector 28 appropriately controls switch 24. Therefore, one of cells C1 and C2 (which is designated by controller 16 and in which the information is stored) at an address (i.e., a register) of the register memory location to be accessed is connected to data read bus 20, so that the information is read out and transferred to main memory 14 through data bus 21 and gate circuit 30.

Write register selector 26 is connected to read register selector 28 through date bus 32. Stored contents of flag bits 13a and 13b of register Ri of register unit 12 are supplied through data lines 34 and 36. Execution mode flag MF is also supplied to selectors 26 and 28. Execution mode flag MF is 1-bit binary data which represents whether a call mode such as a subroutine call or interruption or a return mode such as subroutine return or interruption return of a software program algorithm is currently executed.

Write register selector 26 receives data write request CWR and data write address data (for designating an address of a register memory at which data is to be written) CWA from CPU 10. Selector 26 also receives data write request PWR and data write address data (for designating one of front and back cells C1 and C2 of the address of the register memory at which the data is to be written) PWA from controller 16. Selector 26 controls switch 22 so that one of cells C1 and C2 which is determined on the basis of change flags SB and WB of each cell, execution mode flag MF, and data CWR, CWA, PWR, and PWA supplied from CPU 10 and controller 16 is properly connected to data write bus 18.

Read register selector 28 receives data read request CRR and data read address data (for designating an address of a register memory from which data is to be read out) CRA from CPU 10. Selector 28 receives data read request PRR and data read address data (for designating one of front and back cells C1 and C2 at the address of the register memory from which the data is to be read out) PRA. Selector 28 controls switch 24 so that one of cells C1 and C2 which is determined on the basis of change flags SB and WB of each cell, execution mode flag MF, and data CRR, CRA, PRR, and PRA supplied from CPU 10 and controller 16 is properly connected to data write bus 20.

As shown in Fig. 1, register save/restore controller 16 is connected to CPU 10, data write bus 18, data read bus 20, write register selector 26, and read register selector 28. Controller 16 includes first, second, and third counters 40, 42, and 44. First counter 40 serves as a save/restore register number counter for designating a desired register to be saved/restored by in accordance with an address of the register. Second counter 42 serves as a maximum register number counter for counting a maximum register cell number for defining a maximum register memory amount which can be saved/restored at the same time in register unit 12. Third counter 44 serves as a save/restore register number counter for designating an address of a memory location of main memory 14 to which stored contents currently being stored in register unit 12 are saved or designating an address of the memory location of main memory 14 at which data to be restored to register unit 12 is stored. Controller 16 also includes a register (to be referred to as a "mode flag register" hereinafter) 46 for storing flag MF described above for specifying an operation mode (i.e., either of the call and return modes) currently being executed in the micro-processor.

When CPU 10 receives instructions such as a subroutine call or software interruption while executing software programs, controller 16 is set up as follows by CPU 10 through save/restore setup bus 50 connected to data write bus 18 and data bus 42:

(1) contents of counter 40 are reset to "0";

(2) contents of counter 42 are reset to the number of one or a plurality of registers to be saved of register unit 12;

(3) contents of counter 44 are set to a first address number of a memory area of main memory 14 in which stored contents of register unit 12 are to be saved; and

(4) contents of mode flag register 46 are set to "0" which represents the call mode (therefore, execution mode flag MF of logic value "0" is supplied to write and read register selectors 26 and 28). As a result, controller 16 saves register contents of register areas of register unit 12 from register address number "0" (i.e., register R0) to maximum register address number "n" (i.e., register Rn) to main memory 14 while sequentially updating a memory address counter value by "1". In this case, CPU status data CS is continuously supplied from CPU 10 to controller 16 through data bus 54. Data CS represents a current operation state of CPU 10. On the basis of data CS, controller 16 controls write and read register selectors 26 and 28 so that data is saved from register unit 12 to main memory 14 only in a cycle in which no data is transferred between CPU 10 and main memory 14 (i.e., a data transfer empty cycle which is known as an "empty memory bus cycle" to those skilled in the art). This empty memory bus cycle is set in CPU 10 when, for example:

CPU 10 is transferring operands between registers; and

an instruction pre-fetching buffer is braked upon input of a jump instruction (in this case, an operation of CPU 10 is stopped until CPU 10 fetches a jump destination instruction).

When data saved to main memory 14 is returned (restored) to register unit 12, CPU 10 executes instructions such as subroutine return or interruption return, and controller 16 is set up as follows by CPU 10:

(1) contents of counter 40 are reset to "0";

(2) contents of counter 42 are reset to the number of one or a plurality of registers to be restored of register unit 12;

(3) contents of counter 44 are set to a first address number of a memory area of main memory 14 in which data to be restored of register unit 12 is stored; and

(4) contents of mode flag register 46 are set to "1" which represents the return mode (therefore, execution mode flag MF of logic value "1" is supplied to write and read register selectors 26 and 28). As a result, controller 16 transfers the data to be restored of register areas of register unit 12 from register address number "0" (i.e., register R0) to maximum register address number "n" (i.e., register Rn) to main memory 14 while sequentially updating a memory address counter value by "1". In this case, CPU status data CS is continuously supplied from CPU 10 to controller 16 through data bus 54. Data CS represents a current operation state of CPU 10. On the basis of data CS, controller 16 controls write and read register selectors 26 and 28 so that data is restored from main memory 14 to register unit 12 only in an empty memory bus cycle.

Data to be written in main memory 14 is transferred to main memory 14 through data write bus 18. Data read out from main memory 14 is supplied to register unit 12 through data read bus 20. Gate circuit 30 is provided between main memory 14 and bus 20. Gate circuit 30 opens to connect main memory 14 to bus 20 when stored contents of register unit 12 are to be saved to main memory 14; and closes to connect main memory 14 to bus 18 when stored data of main memory 14 is restored to register unit 12.

Gate circuit 30 is connected to selector 60. Selector 60 selects either memory read/write signal CRW which is generated by CPU 10 or memory read/write signal PRW which is generated by controller 16, and transfers the selected signal to gate circuit 30 and main memory 14 in response to save/restore request signal PR from controller 16. Selector 60 also selects either memory address data CMA which is prepared by CPU 10 or memory address data PMA which is prepared by controller 16, and supplies the selected data to gate circuit 30 and main memory 14 in response to save/restore request signal PR.

In the circuit arrangement described above, one of most important characteristics is such that each register of register unit 12 is constituted by a pair of front and back cells C1 and C2 in the same address. This register cell arrangement is essential to save/restore data between register unit 12 and main memory 14 independently of soft program execution cycles of CPU 10 and utilizing empty time intervals between the execution cycles. That is, while a given process is executed by CPU 10 using front cell C1, register information for another process is allowed to be protected in back cell C2. As a result, old register information need not be saved to main memory 14 when these processes are switched. If three or more of processes are to be executed, stored contents of one of two cells C1 and C2 currently not being selected are saved to main memory 14 or restored therefrom in each register. Therefore, stored content save/restore processing for register unit 12 does not directly conflict with the process currently being executed by CPU 10. In addition, of two cells C1 and C2 of each register, the remaining cell (e.g., C2) other than the cell (e.g., C1) currently being selected serves as a "back-up cell" for the currently selected cell (e.g., C1). Therefore, when a currently executed process to be executed by CPU 10 is switched to another new process, information for the current process can be kept stored (backed up) in the remaining cell (e.g., C2) other than the currently selected cell (e.g., C1). Therefore, when the information is required again thereafter, (e.g., when register data is referred to with respect to subroutine processing or when an operation must be

returned to an information environment one step before the present, i.e., a choice point one level before the present during execution of identification processing of data and list information in a PROLOG machine, etc.), the backed-up register information for the preceding process can be rapidly supplied to CPU 10.

In order to effectively realize the above function, according to the device for saving and restoring register information of the present invention, selector flag SB and change flag WB are provided to each register in register unit 12. In addition, selection between a pair of cells C1 and C2 which constitute each register of register unit 12 is performed by a switching operation of switches 22 and 24 which are controlled by register selectors 26 and 28 in response to execution flag MF controlled by controller 16.

Execution mode flag MF discriminates the call mode such as a subroutine call or interruption generation from the return mode such as subroutine return or interruption return. Fig. 2 shows a diagram schematically showing how a process executed by CPU 10 is processed by repeating the call and return modes, wherein a plurality of thick white vertical lines represent execution of processing such as instruction execution of a process and thin oblique lines connecting the thick white lines represent execution of a process call or process return (i.e., process switching). In Fig. 2, a process represented by "process 2" is a subprocess which interrupts execution of "process 1" (to be referred to as a "main process" hereinafter) as shown by broken line 70 in Fig. 2.

At a start of the main process (process 1), execution mode flag MF is initialized to logic value "0". This means that the main process starts from the call mode. When an instruction such as a subroutine, interruption, or the like is executed (i.e., when the call mode is set) in a given process, i.e., when a process to be executed is changed from the left to right process in Fig. 2, execution mode flag MF has logic value "0". Therefore, if a plurality of call modes are successively set, execution mode flag MF holds logic value "0". When an instruction such as subroutine return, interruption return, or the like is executed (i.e., when the return mode is set) in the process, i.e., when a process to be executed is returned from the right to left process in Fig. 2, execution mode flag MF has logic value "1". Therefore, if a plurality of the return modes are successively set, execution mode flag MF holds logic value "1". When the above event occurs, change flag WB is reset to have a logic value "0". Meanings of the three main flags are summarized in the following table.

	Logic Value "0"	Logic Value "1"
Selector Flag SB	Cell C1 Is Selected	Cell C2 Is Selected
Change Flag WB	Register Contents Are Not Rewritten (Flag Is Reset When Event Occurs And Selected Cell Is Changed)	Register Contents Are Rewritten (New Information Is Written In New Selected Cell And Register Contents Before Event Are Protected In Back-up Cell)
Execution Mode Flag MF	CPU 10 Is In Call Mode	CPU 10 Is In Return Mode

An operation (algorithm) of the device for saving and restoring register information having the above arrangement according to the present invention will be described in detail below. The operation will be described with reference to two states having different logic values of execution mode flag MF which is controlled by save/restore controller 16, i.e., with reference to the call mode and the return mode of CPU 10.

Call Mode

When CPU 10 encounters an event such as a subroutine or interruption while executing a given process, i.e., when CPU 10 executes an instruction in the call mode, register contents (register information) for defining an information environment of the process currently being executed must be saved to a "back-up cell" of a corresponding register of register unit 12 or to a proper address of a memory location of main memory 14. In this case, register accessing by CPU 10 is performed as follows.

Fig. 3A shows register Ri which is one of registers of register unit 12, wherein, for example, selector flag SB has logic value "1" (therefore, back cell C2 is selected) and change flag WB has logic value "0". At this time, switch 24 is switched as shown in Fig. 3A under the control of read register selector 28, and back cell C2 is connected to data read bus 20. Therefore, current process information stored in cell C2 are read out as represented by arrow 30. That is, the register data is always read out firstly from a selected cell (in this case, cell C2) of the two cells of register Ri.

Information is written in register Ri in two ways in accordance with a logic value of change flag WB. That is, when selector flag SB has logic value "0" and change flag WB has logic value "0", selector flag SB is inverted to have logic value "1", and switch 22 is switched under the control of write register selector 26 to write the information in a cell represented by inverted selector flag SB, i.e., in this case, cell C1. At this time, change flag WB is set to have logic value "1" as indicated by arrow 84. Therefore, the information is written in cell C2 as indicated by arrow 86. On the other hand, when change flag WB has logic value "1", selector flag SB is not inverted, and switch 22 is switched to write the information in a cell designated by selector flag SB having logic value "1", i.e., cell C2, as shown in Fig. 3C. At this time, change flag WB holds logic value "1".

By performing the above register reading/writing operation, after CPU 10 completes processing such as subroutine or interruption, register information stored before the processing is performed can be immediately accessed.

It should be noted that, since change flag WB is automatically reset to logic value "0" when an event such as a subroutine call is executed, the first register writing operation in the subroutine (or interruption) is performed under the condition that change flag WB = 0. In this case, as shown in Fig. 4, selector flag SB is inverted, and information "Y" to be written is stored in a register cell which is designated by inverted selector flag SB as indicated by arrow 88 of Fig. 4. At this time, change flag WB is set to logic value "1". By this procedure, while the register cell information stored in one cell (in this case, front cell C1 as shown in Fig. 4) of the register of register unit 12 before occurrence of the event is held therein (i.e., cell C1), new information to be written can be stored in the other cell (i.e., back cell C2) of the register. In other words, when an event such as a subroutine call occurs for the first time in the call mode of CPU 10, register information stored in a cell immediately before occurrence of the event is not erased but is successively held therein, and the other cell (which serves as a "back-up cell") of the register is selected so that the new information is written therein. Therefore, since the register cell information stored immediately before occurrence of the event need not be saved to main memory 14, an environment of the register information concerning processing of the event can be rapidly changed.

A data read algorithm of a register according to register saving in the call mode will be described below. When an event such as a subroutine call occurs while CPU 10 executes a given process, CPU 10 performs set-up processing of controller 16 as described above so as to save register contents relating to the above process executed immediately before occurrence of the event to a suitable address of the memory location of main memory 14. The register information to be saved is read out from register unit 12 in two ways in accordance with a logic value of change flag WB of corresponding register Ri. When change flag WB has logic value "0" (see Fig. 5A), i.e., when stored contents of a cell currently being selected are not rewritten (in this case, register contents in a cell at the subroutine side are not rewritten), information stored in a cell designated by selector flag SB is read out. In Fig. 5A, since selector flag SB has logic value "0", contents "X" of front cell C1 are read out as indicated by arrow 90. When change flag WB has logic value "1" (see Fig. 5B), i.e., when stored contents of a cell currently being selected are rewritten, information stored in a cell opposite to a cell designated by selector flag SB is read out. This is because, in this case, the register contents in the cell at the subroutine side are rewritten and therefore information to be saved is present in the other cell (i.e., a "back-up cell"). In Fig. 5B, since selector flag SB has logic value "1", contents "Y" of front cell C1 are read out as indicated by arrow 92.

Assume that in the call mode, an event such as another subroutine call, subroutine return and interruption occurs before stored contents of a necessary number of registers of register unit 12 are saved to main memory 14. In this case, occurrence of the above event can be detected by monitoring information on monitor bus 52 shown in Fig. 1. When the event such as a subroutine call or interruption occurs,

execution of this event is temporarily delayed, and saving processing of the above register contents is executed prior to execution of the event. This is to prevent destruction of register information to be saved when information of the event is stored in register unit 12. On the contrary, when the event is subroutine return, the register content saving processing is immediately stopped (interrupted), and the event is executed prior to the processing. This is because, in this case, the previous register information required for executing the above subroutine return is stored not in a currently selected cell but in the other cell serving as a "back-up cell" for the currently selected cell of two cells in each of the corresponding registers.

Return Mode

When CPU 10 encounters an event such as subroutine return while executing a given process, i.e., when CPU 10 executes an instruction in a return mode, register information for defining an information environment of the preceding process which is saved to main memory 14 must be restored to register unit 12. In this state, CPU 10 performs register accessing as follows.

Register read in the return mode is performed such that a cell designated by selector flag SB of register Ri is subjected to a reading operation, as in the call mode. That is, in Fig. 6A, for example, since selector flag SB has logic value "1", back cell C2 is currently selected. Therefore, register information of cell C2 is read out as indicated by arrow 94.

Information write into register Ri is performed with respect to a cell designated by selector flag SB in the same manner as described above. In Fig. 6B, for example, since selector flag SB has logic value "1", switch 22 is switched so that information to be written is supplied to cell C2 as indicated by arrow 96. In this case, a logic value of change flag WB is not changed (set up).

In the return mode, in order to prepare for another subroutine return which is expected to occur next to the return event currently being executed, register information restoring processing must be performed to a non-selected cell other than the currently selected cell. In order to perform this processing, after CPU 10 executes one subroutine, it sets up change flag WB of corresponding register Ri to prepare for the next return event. For example, according to Fig. 7, a logic value of change flag WB is changed from "0" to "1". Therefore, each time an empty cycle is generated in data transfer bus 18 or 20 of CPU 10 (this empty cycle is generated when, e.g., (a) CPU 10 executes a data transfer instruction between registers, (b) an instruction pre-read register is invalidated or valve-broken while a jump instruction is executed, and (c) an instruction cache memory is provided and a desired instruction is not hit in the cache memory), register information to be restored is read out from main memory 14 and is written in a corresponding non-selected cell (i.e., a cell opposite to the cell currently being designated by selector flag SB and serves as a "back-up cell") of register unit 12. In Fig. 7, since selector flag SB has logic value "1", the register information to be restored is written in cell C1 opposite to cell C2 which is designated by selector flag SB as indicated by arrow 98. When this information write is completed, a logic value of change flag WB is changed from "0" to "1" as indicated by arrow 100.

When CPU 10 executes an instruction of the next subroutine return after restore processing of the register information is completed, selector flags SB of registers whose change flags WB have logic values "1" of register unit 12 are automatically inverted. At this time, change flags WB having logic values "1" are all reset to logic value "0". As a result, all change flags WB have logic values "0". Therefore, in this state, selector flag SB in register Ri represents a cell in which the register information is restored.

Assume that in the return mode, an event such as another subroutine call, subroutine return, or interruption occurs before register information is completely restored to a necessary number of registers of register unit 12. In this case, if a call mode event such as a subroutine call or interruption occurs, the register restore processing is immediately stopped (interrupted), and the event is executed prior to the processing. On the contrary, if the event is a subroutine return, execution of the event is temporarily delayed, and register content save processing is executed prior to the event.

According to the device for saving and restoring register information, a pair of register cells C1 and C2 having the same register address is provided to each register of register unit 12. Therefore, when register contents of register Ri are to be changed, an amount of information transfer for saving/restoring between register unit 12 and main memory 14 can be reduced. As a result, save/restore processing of register unit 12 can be efficiently performed.

More specifically, register contents of register Ri of register unit 12 must be rewritten with another new information in various kinds of events, such as process switching of a micro-processor adapted to a specific operation system, identification processing of a micro-processor specified to a PROLOG machine, or various events such as a subroutine call, interruption, subroutine return, and interruption return which occur

during execution of a software program. In such a case, the above new information is written in a register cell (e.g., cell C2) other than a register cell (e.g., cell C1) currently being selected using selector flag SB and change flag WB provided to register Ri and execution mode flag MF controlled by save/restore controller 16. At this time, cell C2 stores the new information as a newly selected cell, and the old register information is kept protected in cell C1 serving as a "back-up cell". Since the old register information is still present in register unit 12 in register content change of one level as described above, save processing to main memory 14 need not be performed for register Ri. When the old register information is required again, this information can be restored in register Ri by only selecting "back-up cell" C2 again. This back-up protection of information is performed independently of other registers in units of registers in register unit 12. Therefore, when register contents of register unit 12 are to be partially changed frequently, register unit 12 can be accessed at a high speed.

Register information of register Ri must be saved to main memory 14 only when still another information must be subsequently written in register Ri (i.e., register content change of two levels or more). Since this save processing (although restore processing is similar to the save processing) is performed utilizing empty cycles of CPU 10 (this cycle is essentially a useless time for CPU 10), a processing speed of CPU 10 is not basically reduced. Actually, according to experiments conducted by the present inventors, it is confirmed that the empty cycles are generated so frequently as to cancel a time required for register saving/restoring. Therefore, in this case, a time consumed for register saving/restoring is actually zero.

The above characteristics greatly contribute to save/restore register contents of register unit 12 in process switching, identification processing of a PROLOG machine, various events which occur during execution of a software program within minimum reduced time at a high speed. Therefore, the above characteristics are very important because an actual operation speed of a micro-processor can be improved without changing any basic specifications. In addition, old register information one level before register information currently being used is kept stored in register unit 12. Therefore, an information environment one level before the present can be immediately activated as needed in, e.g., the PROLOG machine, thereby increasing a processing speed of an artificial intelligence task.

Furthermore, according to the present invention, since selector flag SB and change flag WB are provided to each register, the states of two cells C1 and C2 of a designated register can be clearly recognized. Therefore, when register contents are to be changed or register saving/storing is to be executed, processing for determining which cell of cells C1 and C2 of register Ri is to be accessed can be properly and accurately executed on the basis of selector flag SB, change flag WB, and execution mode flag MF which is controlled by save/restore controller 16. This characteristic contributes to maintain consistency of register cell accessing of register unit 12.

Claims

1. A register device for a central processing unit of a computer, characterized by comprising: a register unit (12) having registers each of which has a pair of register cells of first and second cells (C1, C2), and when one of said first and second cells is selected, the other cell serving as a back-up cell for said selected cell; first flag holding means (13a), provided to each register, for holding therein a first flag (SB) representing which cell of said first and second cells (C1, C2) of said register is currently selected; second flag holding means (13b), provided to each register, for holding therein a second flag (WB) representing whether register information of said register is rewritten after a selected cell is changed between said first and second cells of said register; and controller means (16, 22, 24, 26, 28), connected to said register unit (12), said first flag holding means (13a), and said second flag holding means (13b), for, when first register information is to be written in a given register of said register unit, selecting one of said first and second cells of said register into which the first register information is stored, wherein, the other cell of said first and second cells remains as a non-selected cell, and for, when the first register information is to be rewritten with second register information in said register, selecting said other cell and storing the second register information therein and at the same time holding the first register information in said one cell, whereby protecting the first register information in the register unit after the first register information is rewritten with the second register information.

2. The device according to claim 1, characterized in that said controller means comprises switching means (22, 24), connected to said register unit, for independently switching a selected cell between said first and second cells of each of said registers of said register unit and supplying an information transfer path to a cell currently being selected in each register.

3. The device according to claim 2, characterized by further comprising a main memory (14), and wherein when third register information is to be written in said register unit (12), said controller means detects a cell of said register currently being selected and a storage state of said cell on the basis of said first and second flags (SB, WB), saves the first register information held in said one cell currently not being selected to said main memory (14), switches a selected cell between said first and second cells (C1, C2) so as to select said one cell again, and stores the third register information in said one cell, the second register information being held in said other cell which is not selected.

4. The device according to claim 3, characterized in that said first flag (SB) has a binary bit which is inverted each time a selected cell is changed in a corresponding register, and wherein said second flag (WB) has a binary bit which is inverted when information currently being used in said register is changed from the first to second register information.

5. A microprocessor comprising a central processing unit (10), and a main memory (14), characterized in that said microprocessor further comprises a register unit (13b) having registers (R), each having a pair of cells consisting of first and second cells (C1, C2) having the same register address, a first flag storage portion (13a) for storing a selector flag (SB) of binary bit data for representing which cell of said first and second cells is currently selected, and a second flag storage portion (13b) for storing a change flag (WB) of binary bit data for representing whether information stored in a selected cell is rewritten with another new information in a corresponding register; a first data bus (18) for transferring information to be written in said register unit between said central processing unit and said register unit; a second data bus (20) for transferring information read out from said register unit between said central processing unit and said register unit; a third data bus (21) for allowing information transfer between said main memory and said first and second data buses; first switching means (22), provided between said register unit and said first data bus, for selectively connecting one of said pair of cells of each register to said first data bus; second switching means (24), provided between said register unit and said second data bus, for selectively connecting one of said pair of cells of each register to said second data bus; and register control means (16, 26, 28), connected to said central processing unit and said first and second switching means, for, when stored contents of a first cell of said pair of cells of a certain register of said register unit designated as a selected cell by said selector flag are to be changed to new information, independently executing save processing of the stored contents of said selected cell on the basis of said selector and change flags in accordance with a manner comprising the steps of inverting said selector flag so that said selected cell in said register is changed from said first cell to said second cell, switching said first switching means so that said second cell is connected to said first data bus to, thereby store the new information in said second cell, and holding stored contents of said first cell as a non-selected cell therein to eliminate necessity of saving the stored contents of said first cell to said main memory at this stage.

6. The microprocessor according to claim 5, characterized in that when stored contents of said first cell (C1) are required again, said control means inverts said selector flag (SB) of said certain register again so as to return said selected cell to said first cell and switches said second switching means (24) so as to connect said first cell to said second data bus (20), so that the stored contents of said first cell are accessed by said central processing unit.

7. The microprocessor according to claim 5, characterized in that when another new information must be stored in said certain register, said control means switches said second switching means (24) so that said first cell of said certain register is connected to said second data bus (20), thereby saving the stored contents of said first cell in said main memory (14) through said second and third data buses (20, 21), and wherein said control means inverts said selector flag (SB) to select said first cell, and switches said first switching means (22) to connect said first cell to said first data bus (18), thereby storing the another new information in said first cell.

8. The microprocessor according to claim 7, characterized in that said control means saves the stored contents of said first cell to said main memory (14) in empty cycles generated during operation of said central processing unit.

9. The microprocessor according to claim 8, characterized in that when the contents saved to said main memory (14) are to be restored in said certain register, said control means switches said first switching means (22) so that said second cell, which is a cell opposite to a cell currently being designated by said selector flag and remains as a non-selected cell, is connected to said first data bus which is connected to said main memory through said third data bus (21), thereby restoring the contents saved to said main memory in said second cell through said first and third data buses (18, 21), and wherein said control means inverts said selector flag so that said second cell is designated as a selected cell in said certain register.

10. A microprocessor comprising a central processing unit (10), and a main memory (14), characterized in that said microprocessor further comprises a register unit (12) having registers (R), each having a pair of cells consisting of first and second cells (C1, C2) having the same address, a first flag storage portion (13a) for storing a selector flag (SB) of binary bit data for designating which cell of said first and second cells is currently designated, and a second flag storage portion (13b) for storing a change flag (WB) of binary bit data which is set to have a precetermined logic value opposite to an initial logic value when information stored in a selected cell of a corresponding register is rewritten with another new information; data bus means (18, 20, 21) for allowing information transfer between said central processing unit, said register unit, and said main memory; first switching means (22), provided between said register unit and said data bus means, for selectively connecting one of said pair of cells of each register to said data bus means and supplying register information to be written in said register thereto; second switching means (24), provided between saic register unit and said data bus means, for selectively connecting one of said pair of cells of each register to said data bus means and supplying register information from said cell to said data bus means; and register control means (16, 26, 28), connected to said central processing unit and said first and second switching means, for generating an execution mode flag (MF) of binary bit data for representing in which mode of call and return modes said central processing unit currently operates, and when in the call mode, said change flag is reset to the initial logic value and stored contents of a first cell currently being designated as a selected cell by said selector flag of said pair of cells of a certain register of said register unit are to be changed to new information, for independently executing save processing of the stored contents of said first cell of each register in accordance with a manner comprising the steps of inverting said selector flag so as to select said second cell in said certain register, switching said first switching means so that said second cell is connected to said data bus means to thereby store the new information in said second cell, setting said change flag to have the predetermined logic value, and holding stored contents of said first cell as a back-up cell therein, thereby eliminating necessity of saving the stored contents of said first cell to said main memory at this stage.

11. The microprocessor according to claim 10, characterized in that when the stored contents of said first cell (C1) are required again in the return mode of said central processing unit, said control means inverts said selector flag (SB) again to select said first cell in said certain register and switches said second switching means (24) to connect said first cell to said data bus means (18, 20, 21), so that the stored contents of said first cell can be accessed by said central processing unit.

12. The microprocessor according to claim 10, characterized in that when another new information must be stored in said certain register in the call mode of said central processing unit, said control means executes save processing of said certain register on the basis of said selector and change flags (SB, WB) in accordance with a manner comprising the steps of

switching said second switching means (24) to connect said first cell, which is a cell opposite to said second cell (C2) currently being designated by said selector flag, of said certain register to said data bus means (18, 20, 21), thereby saving the stored contents of said first cell to said main memory (14),

switching said first switching means (22) to connect said first cell to said data bus means, thereby storing the another new information in said first cell, and

inverting said selector flag (SB) to select said first cell (C1).

13. The microprocessor according to claim 12, characterized in that control means saves the stored contents of said first cell (C1) to said main memory (14) in empty cycles generated during operation of said central processing unit.

14. The microprocessor according to claim 13, characterized in that when the contents saved to said main memory (14) are restored in said certain register in the return mode of said central processing unit, said control means executes restore processing in accordance with a manner comprising the steps of switching said first switching means (22) to connect said second cell of said certain register to said data bus means (18, 20, 21) to thereby restore the saved contents from said main memory to said second cell which is a cell opposite to a cell currently being designated by said selector flag, and inverting said selector flag (SB) to select said second cell (C2).

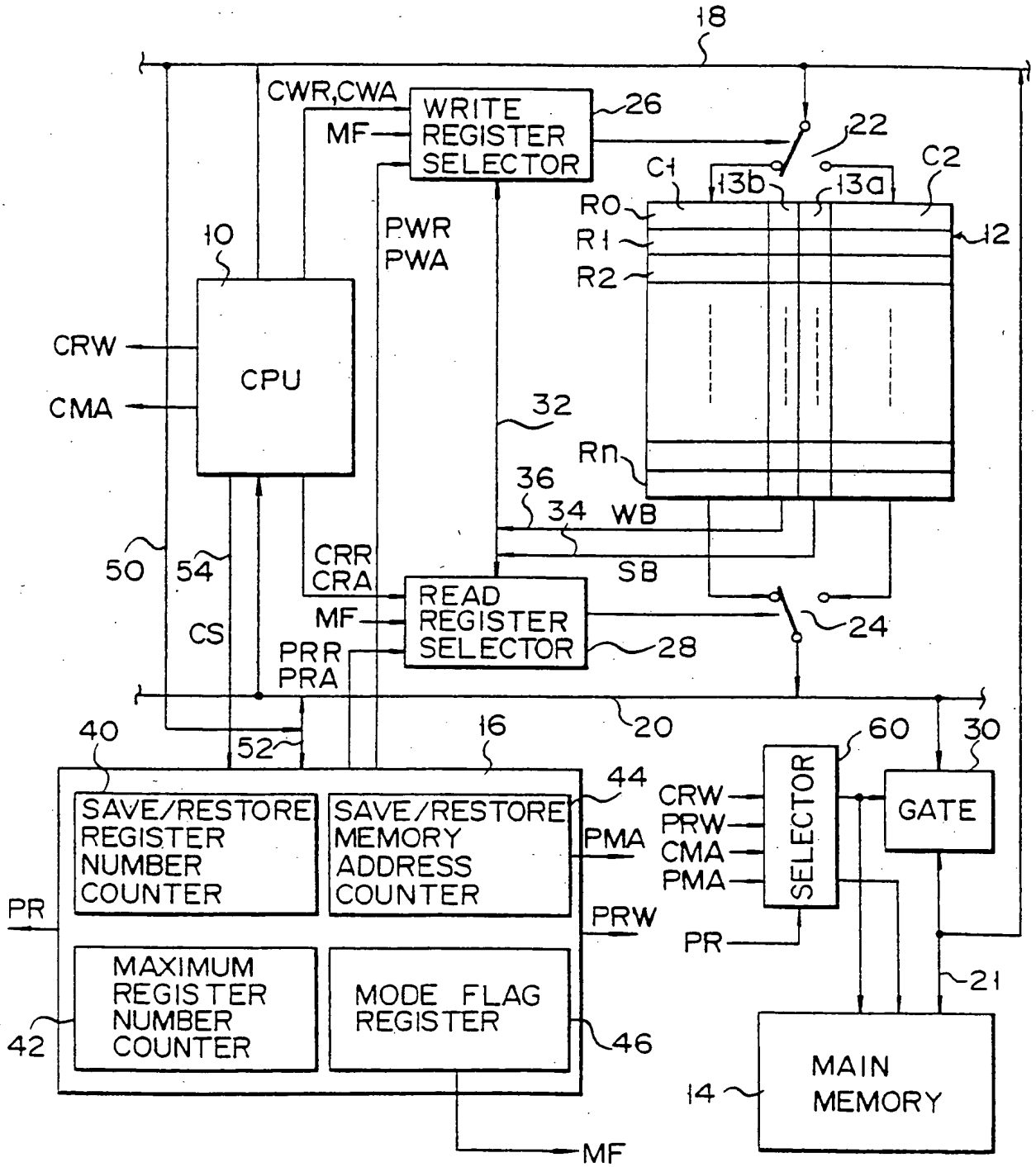
15. The microprocessor according to claim 14, characterized in that said control means restores the saved contents from said main memory (14) to said second cell (C2) in empty cycles generated during operation of said central processing unit.

16. The microprocessor according to claim 15, characterized in that said control means comprises a flag register for storing the execution mode flag (MF).

17. The microprocessor according to claim 16, characterized in that said control means comprises first counter means (40) for sequentially designating addresses of specific registers including said certain register to be subjected to save-restore processing, and second counter means (44) for defining addresses of said main memory (14) to which stored contents of registers to be subjected to save processing are to be saved and defining addresses of said main memory to be subjected to restore processing.

18. The microprocessor according to claim 17, characterized in that said control means comprises third counter means (42) for storing the number of said specific registers to be subjected to the save-restore processing.

19. The microprocessor according to claim 18, characterized in that said data bus means comprises a first data bus (18) for transferring information to be written in said register unit (12) between said central processing unit and said register unit, a second data bus (20) for transferring information read out from said register unit between said central processing unit and said register unit, and a third data bus (21) for allowing transfer of information between said main memory (14) and said first and second data buses.


18
CWR,CWA
MF
WRITE REGISTER SELECTOR
26
22
C1
13b
13a
C2
R0
R1
R2
12
10
PWR
PWA
CRW
CMA
CPU
32
Rn
36
34
WB
SB
50
54
CRR
CRA
MF
READ REGISTER SELECTOR
28
24
CS
PRR
PRA
20
40
52
16
60
30
44
SAVE/RESTORE REGISTER NUMBER COUNTER
SAVE/RESTORE MEMORY ADDRESS COUNTER
PMA
CRW
PRW
CMA
PMA
SELECTOR
GATE
PR
PRW
PR
21
MAXIMUM REGISTER NUMBER COUNTER
MODE FLAG REGISTER
42
46
14
MAIN MEMORY
MF

F I G. 1

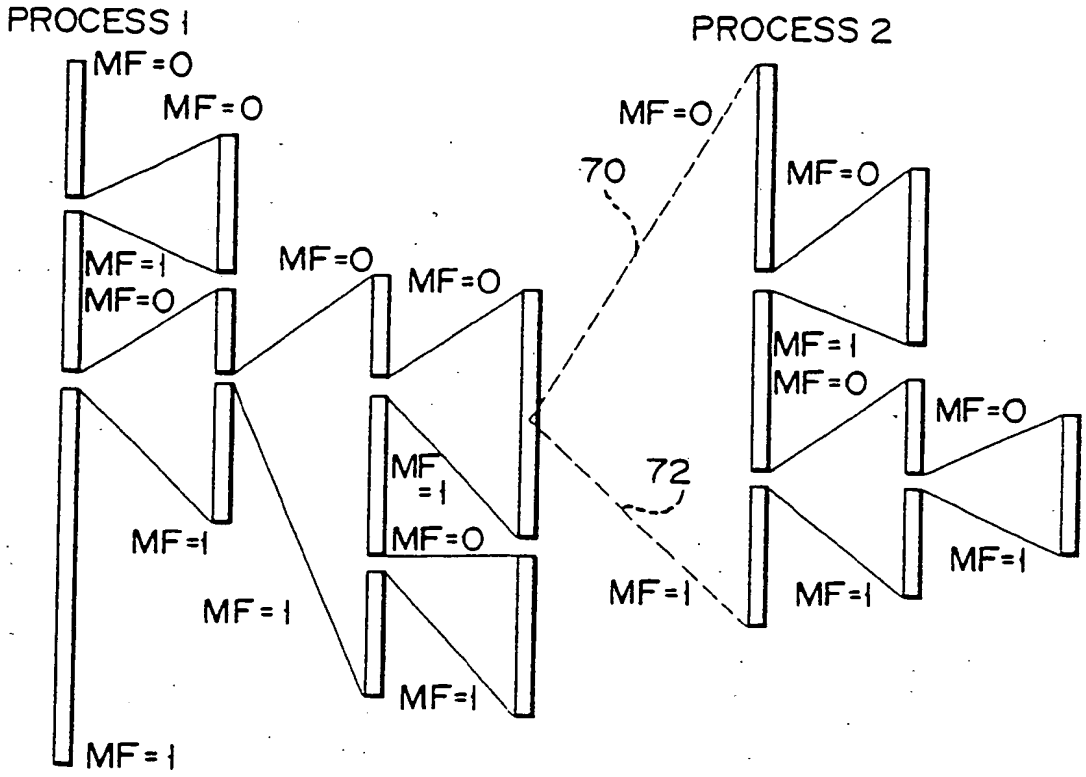

PROCESS 1
PROCESS 2
MF=0
MF=0
MF=1
MF=0
MF=0
MF=0
MF=1
MF=1
MF=1
MF=1
MF=1
MF=0
70
72
MF=0
MF=1
MF=0
MF=0
MF=1
MF=1
MF=1

FIG. 2

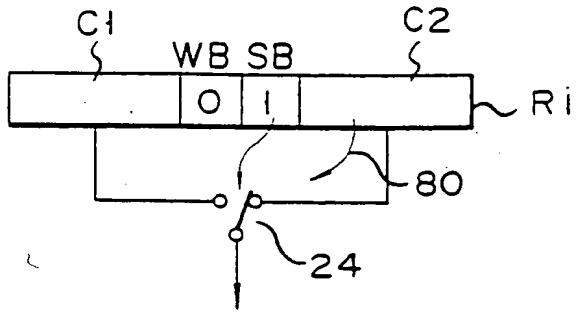


FIG. 3A

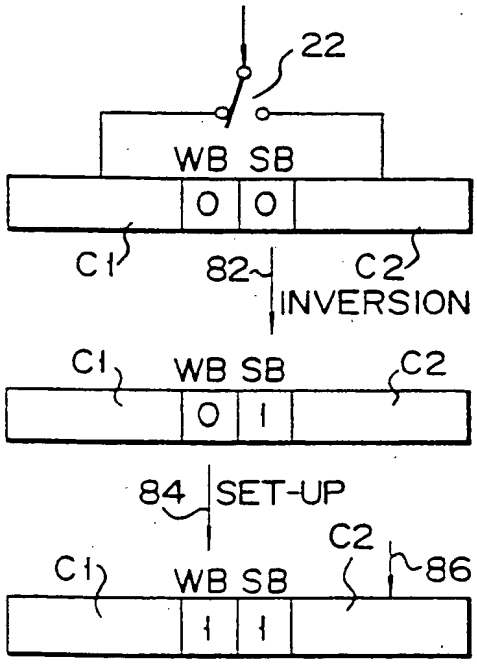


FIG. 3B

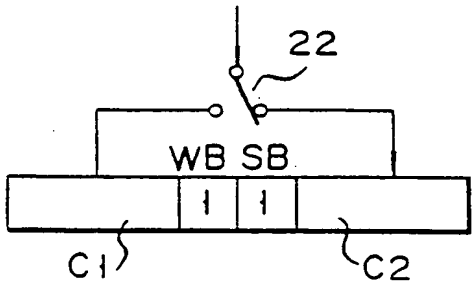


FIG. 3C

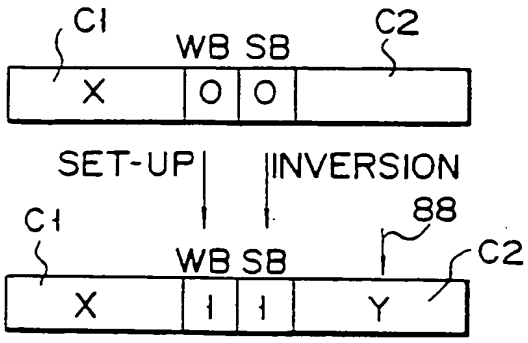


FIG. 4

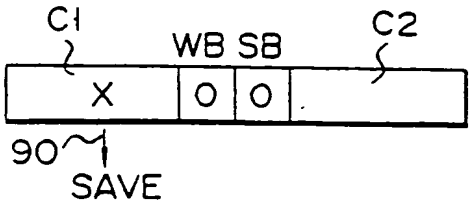


FIG. 5A

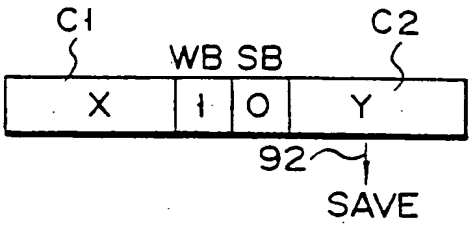


FIG. 5B

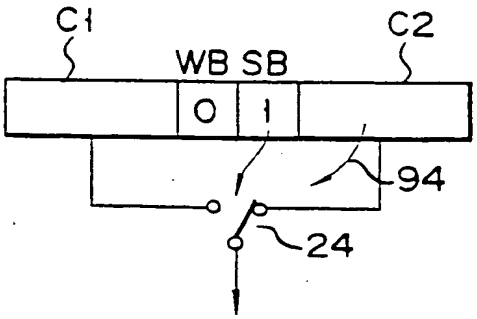


FIG. 6A

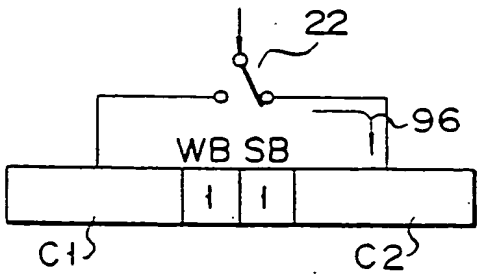


FIG. 6B

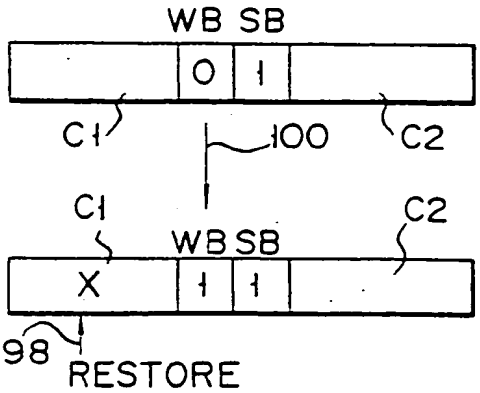


FIG. 7

THIS PAGE BLANK (USPTO)

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: 0 285 310 A3

(12) EUROPEAN PATENT APPLICATION

(21) Application number: 88302516.5

(51) Int. Cl.[5]: G06F 9/46

(22) Date of filing: 22.03.88

(30) Priority: 31.03.87 JP 78506/87

(43) Date of publication of application:
05.10.88 Bulletin 88/40

(84) Designated Contracting States:
DE FR GB

(88) Date of deferred publication of the search report:
16.08.90 Bulletin 90/33

(71) Applicant: KABUSHIKI KAISHA TOSHIBA
72, Horikawa-cho Saiwai-ku
Kawasaki-shi Kanagawa-ken 210(JP)

(72) Inventor: Matoba, Tsukasa c/o Patent Division
Kabushiki Kaisha Toshiba 1-1 Shibaura 1-chome
Minato-ku Tokyo 105(JP)
Inventor: Aikawa, Takeshi c/o Patent Division
Kabushiki Kaisha Toshiba 1-1 Shibaura 1-chome
Minato-ku Tokyo 105(JP)
Inventor: Okamura, Mitsuyoshi c/o Patent Division
Kabushiki Kaisha Toshiba 1-1 Shibaura 1-chome
Minato-ku Tokyo 105(JP)
Inventor: Maeda, Ken-ichi c/o Patent Division
Kabushiki Kaisha Toshiba 1-1 Shibaura 1-chome
Minato-ku Tokyo 105(JP)
Inventor: Saito, Mitsuo c/o Patent Division
Kabushiki Kaisha Toshiba 1-1 Shibaura 1-chome
Minato-ku Tokyo 105(JP)

(74) Representative: Freed, Arthur Woolf et al
MARKS & CLERK 57-60 Lincoln's Inn Fields
London WC2A 3LS(GB)

(54) Device for saving and restoring register information.

(57) Each register of an internal register unit (12) of a microprocessor has a pair of register cells consisting of first and second cells (C1, C2) having the same register address. When one of these cells is selected, the other cell non-selected serves as a "back-up cell" for the selected cell. Each register of the register unit (12) has a flag bit (13a) for storing a selector flag (SB) representing which cell of the pair of cells of the register is currently selected, and a flag bit (13b) for storing a change flag (WB) representing whether register information of the register is rewritten after a selected cell is changed between the first and second cells (C1, C2) of the register. When the register information is stored in one of the pair of cells currently being selected of a certain register and is to be rewritten with another new information, the other cell of the register is selected to store the new information therein. The original register information is held in the first cell (C1), thereby eliminating necessity of saving the original register information to a main memory (14) at this stage. When the original register information is required again, the original register information can be rapidly restored, by only selecting the first cell again, in the corresponding register without executing save/restore processing between the register unit (12) and the main memory (14).

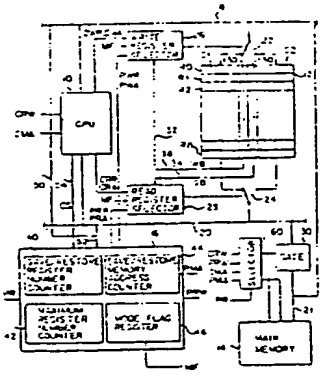


FIG. 1

EP 0 285 310 A3

European Patent Office

EUROPEAN SEARCH REPORT

Application Number

EP 88 30 2516

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int. Cl.4)
X	US-A-4 410 939 (KAWAKAMI) * Abstract; column 6, line 54 - column 9, line 34; figures 5,6 *	1-4,10, 11	G 06 F 9/46
Y		5-8,12, 13	
A		9,14	
Y	WO-A-8 302 016 (BURROUGHS) * Abstract; page 3, lines 3-27; page 5, line 32 - page 7, line 25; figures 1,2 *	5-8,12, 13	
A		10	
Y	US-A-4 181 938 (SUZUKI) * Abstract; column 3, line 66 - column 4, line 6; figure 3 *	8,13	

TECHNICAL FIELDS SEARCHED (Int. Cl.4)

G 06 F 9/46
G 06 F 9/42
G 06 F 5/06

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	17-05-1990	DHEERE R.F.B.M.

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P0401)